

About.java

```
package com.taksijogja.app;
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
public class About extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_about); }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if
it is present.
        getMenuInflater().inflate(R.menu.about, menu);
        return true;
    }
}
```

databaseHelper.java

```
package com.taksijogja.app;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
public class DatabaseHelper extends SQLiteOpenHelper{
    public DatabaseHelper(Context context) {
        super(context, "taksi.db", null, 1);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        // TODO Auto-generated method stub
    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion,
int newVersion) {
        // TODO Auto-generated method stub
    }
}
```

Koordinat.java

```
package com.taksijogja.app;
public class Koordinat {
    public static String KEY_IDKOORDINAT = "id_kordinat";
    public static String KEY_IDPANGKALAN = "id_pangkalan";
    public static String KEY_LATITUDE = "latitude";
    public static String KEY_LONGITUDE = "longitude";
    private int id_koordinat;
    private int id_pangkalan;
    private double latitude;
    private double longitude;
    public int getId_koordinat() {
        return id_koordinat;
    }
    public void setId_koordinat(int id_koordinat) {
        this.id_koordinat = id_koordinat;
    }
    public int getId_pangkalan() {
        return id_pangkalan;
    }
    public void setId_pangkalan(int id_pangkalan) {
        this.id_pangkalan = id_pangkalan;
    }
    public double getLatitude() {
        return latitude;
    }
    public void setLatitude(double latitude) {
        this.latitude = latitude;
    }
    public double getLongitude() {
        return longitude;
    }
    public void setLongitude(double longitude) {
        this.longitude = longitude;
    }
}
```

Koordinatimpl.java

```
package com.taksijogja.app;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
public class KoordinatImpl implements SQLiteInterface{
    SQLiteDatabase db;
    DatabaseHelper helper;
    public KoordinatImpl(DatabaseHelper helper){
        this.helper=helper;}
    @Override
    public Cursor getAll() {
        Cursor result = db.query("kordinat", new String[]{
            Koordinat.KEY_IDKOORDINAT,
            Koordinat.KEY_IDPANGKALAN,
            Koordinat.KEY_LATITUDE,
            Koordinat.KEY_LONGITUDE
        }, null, null, null, null, null);
        return result;}
    public DatabaseHelper getHelper
        return helper;}
    public void openKoneksi(){
        db = helper.getWritableDatabase(); }
    public void closeKoneksi(){
        db.close();
        helper.close();
    }
}
```

loadDB.java

```
package com.taksijogja.app;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import android.annotation.SuppressLint;
import android.content.Context;
@SuppressLint("SdCardPath")
public class LoadDB {
    public static final String DB_PATH =
"/data/data/com.taksijogja.app/databases/";
    final String DB_NAME = "taksi.db";
    Context ctx;
    public LoadDB(Context context){
        this.ctx=context;
        try {
            copyDatabase();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
    private void copyDatabase() throws IOException{
        InputStream inputStream =
ctx.getAssets().open(DB_NAME);
        String outfile = DB_PATH+DB_NAME;
        OutputStream out = new FileOutputStream(outfile);
        byte[] buffer = new byte[1024];
        int length;
        while((length=inputStream.read(buffer))>0)
        {
            out.write(buffer,0,length);
        }
        out.flush();
        out.close();
        inputStream.close();
    }
}
```

splash.java

```
package com.taksijogja.app;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;
import android.widget.ProgressBar;
public class Splash extends Activity {
    final int TIMER_RUNTIME = 5000;
    boolean mbActive = false;
    ProgressBar mProgressBar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splash);
        //deleteDatabase("taksi.db");
        new LoadDB(this);
        mProgressBar = (ProgressBar) findViewById(R.id.bar);
        final Thread timerThread = new Thread(){
            @Override
            public void run(){
                mbActive = true;
                try{
                    int waited=0;
                    while(mbActive && (waited<TIMER_RUNTIME)){
                        sleep(20);
                        if(mbActive){
                            waited+=20;
                            updateProgress(waited);}}
                    }catch(InterruptedException e){
                    }finally{
                        onContinue();}}};
                timerThread.start();}
            @Override
            public void onDestroy(){
                super.onDestroy();}
            public void updateProgress(final int timePassed){
                if(null !=mProgressBar){
                    final int progress =
mProgressBar.getMax()*timePassed;
                    mProgressBar.setProgress(progress);}}
            public void onContinue(){
                finish();
                startActivity(new Intent(this,MainActivity.class));}
            @Override
            public boolean onCreateOptionsMenu(Menu menu) {
                // Inflate the menu; this adds items to the action
bar if it is present.
                getMenuInflater().inflate(R.menu.splash, menu);
                return true;}}
```

taksidialer.java

```
package com.taksijogja.app;
public class TaksiDialer {
    private int id;
    private String nama;
    private String alamat;
    private String no_telp;
    private double latitude;
    private double longitude;
    public int getId() {
        return id;
    }
    public void setId(int id) {
        this.id = id;
    }
    public String getNama() {
        return nama;
    }
    public void setNama(String nama) {
        this.nama = nama;
    }
    public String getAlamat() {
        return alamat;
    }
    public void setAlamat(String alamat) {
        this.alamat = alamat;
    }
    public String getNo_telp() {
        return no_telp;
    }
    public void setNo_telp(String no_telp) {
        this.no_telp = no_telp;
    }
    public double getLatitude() {
        return latitude;
    }
    public void setLatitude(double latitude) {
        this.latitude = latitude;
    }
    public double getLongitude() {
        return longitude;
    }
    public void setLongitude(double longitude) {
        this.longitude = longitude;
    }
}
```

Mainactivity.java

```
package com.taksijogja.app;
@SuppressLint("NewApi")
public class MainActivity extends FragmentActivity implements
LocationListener {
    final int RQS_GooglePlayServices = 1;
    private GoogleMap map;
    KoordinatImpl helper;
    private ProgressDialog pDismiss;
    Drawable icon;
    Location location;
    LocationManager locationManager;
    AsyncTaskMain task;
    TaksiDialer selected = null;
    EditText txtCari = null;
    List<TaksiDialer> listTaksi = null;
    Button btnCari;
    String kunci = "";
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
//method ini akan menyeleksi menu yg dipilih pd system.
        switch (item.getItemId()) {
            case R.id.menu_about:
                startActivity(new Intent(this, About.class));
                return true;
            case R.id.menu_refresh:
                map.clear();
                kunci = "";
                setup();
                return true;
            default:
                return true;}}
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        icon =
getResources().getDrawable(R.drawable.icon_taksi);
        helper = new KoordinatImpl(new
DatabaseHelper(this));
        btnCari = (Button) findViewById(R.id.btnCari);
        txtCari = (EditText) findViewById(R.id.txtCari);
        setup();
        btnCari.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                map.clear();
                kunci = txtCari.getText().toString();
```



```

        setup(); //meload hasil peta.
        txtCari.setText("");}}});
    public void setup(){
        listTaksi = new ArrayList<TaksiDialer>();
        FragmentManager myFragmentManager =
getSupportFragmentManager();
        SupportMapFragment mySupportMapFragment =
(SupportMapFragment)
myFragmentManager.findFragmentById(R.id.map);
        map = mySupportMapFragment.getMap();
        map.setMyLocationEnabled(true);
        locationManager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
        Criteria criteria = new Criteria();
        location =
locationManager.getLastKnownLocation(locationManager
        .getBestProvider(criteria, false));
    terakhir.
        task = new AsyncTaskMain();
        task.execute();
        if (location != null) {
            map.animateCamera(CameraUpdateFactory.newLatLngZoom(new
LatLng(
            location.getLatitude(), location.getLongitude()), 13));}
        map.setOnMarkerClickListener(new OnMarkerClickListener()
    {
        @Override
        public boolean onMarkerClick(Marker taksi) {
            AlertDialog.Builder dialog = new
AlertDialog.Builder(MainActivity.this);
            for (int i = 0; i < listTaksi.size(); i++) {
                int d = listTaksi.get(i).getId();
                String nama = listTaksi.get(i).getNama();
                String id = d + "." + nama;
                if (taksi.getTitle().equals(id)) {
                    selected = listTaksi.get(i);
                    break;}}
            dialog.setTitle("" + selected.getNama());
            dialog.setMessage("Alamat : " +
selected.getAlamat());
            dialog.setNeutralButton("CALL",
new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int
which) {
                    // TODO Auto-generated method stub
                    startActivity(new Intent(Intent.ACTION_DIAL,
Uri.parse("tel:"selected.getNo_telp()))));}});
            dialog.setNegativeButton("CANCEL",
new DialogInterface.OnClickListener() {
                @Override

```

```

        public void onClick(DialogInterface dialog,
        int which) {
            // TODO Auto-generated method stub}});
            dialog.show();
            return false;}});}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.about, menu);
    return true;}

private class AsyncTaskMain extends AsyncTask<Void, Void,
Void> {
    @Override
    protected void onPostExecute(Void result) {
        pDismiss.dismiss();
        runOnUiThread(new Runnable() {
            @Override
            public void run() {
                for (int i = 0; i < listTaksi.size(); i++) {
                    Location dest = new Location("");
                    dest.setLatitude(listTaksi.get(i).getLatitude());
                    dest.setLongitude(listTaksi.get(i).getLongitude());
                    double distance = location.distanceTo(dest);
                    if (distance < 3000) {
                        map.addMarker(new MarkerOptions()
                            .title(listTaksi.get(i).getId() +
                                "."+listTaksi.get(i).getNama())
                            .icon(BitmapDescriptorFactory
                                .fromResource(R.drawable.icon_taksi))
                            .position(
                                new LatLng(listTaksi.get(i)
                                    .getLatitude(), listTaksi
                                    .get(i).getLongitude()))); }}}});
                        super.onPostExecute(result);
                    }
                @Override
                protected void onPreExecute(){
                    // TODO Auto-generated method stub
                    super.onPreExecute();
                    pDismiss = new
ProgressDialog(MainActivity.this);
                    pDismiss.setMessage("Loading....");
                    pDismiss.setCancelable(true);
                    pDismiss.show();}

                @Override
                protected Void doInBackground(Void... arg0) {
                    String sql = "";
                    helper.openKoneksi();
                    if(kunci.equals("")){
                        sql = "select td.id_dialer,
                        t.nama_taksi,p.alamat_pangkalan,t.no_telepon,k.latitude,k.long
                        itude "
                        + "from taksi t,taksi_dialer td,pangkalan p,kordinat k "

```

```

        + "where t.id_taksi=td.id_taksi AND
p.id_pangkalan=td.id_pangkalan AND
k.id_pangkalan=td.id_pangkalan;";
    }else{
        sql = "select td.id_dialer,
t.nama_taksi,p.alamat_pangkalan,t.no_telepon,k.latitude,k.long
itude "
        + "from taksi t,taksi_dialer td,pangkalan p,kordinat k "
        + "where t.id_taksi=td.id_taksi AND
p.id_pangkalan=td.id_pangkalan AND
k.id_pangkalan=td.id_pangkalan AND t.nama_taksi
LIKE'" + kunci+"'" ;"; }
    Cursor c = helper.getHelper().getReadableDatabase()
        .rawQuery(sql, null);
    while (c.moveToNext()) {
        TaksiDialer dialer = new TaksiDialer();
        dialer.setId(c.getInt(0));
        dialer.setNama(c.getString(1));
        dialer.setAlamat(c.getString(2));
        dialer.setNo_telp(c.getString(3));
        dialer.setLatitude(c.getDouble(4));
        dialer.setLongitude(c.getDouble(5));
        listTaksi.add(dialer);}
        helper.closeKoneksi();
        return null;}}
    @Override
    public void onLocationChanged(Location newLocation) {
        // TODO Auto-generated method stub
        Criteria criteria = new Criteria();
        location = newLocation;
        location =
locationManager.getLastKnownLocation(locationManager
        .getBestProvider(criteria, false));
        task.execute();
        if (location != null) {

            map.animateCamera(CameraUpdateFactory.newLatLngZoom(new
            LatLng(
                location.getLatitude(), location.getLongitude()),
            13)); }
    @Override
    public void onProviderDisabled(String provider) {
        // TODO Auto-generated method stub }
    @Override
    public void onProviderEnabled(String provider) {
        // TODO Auto-generated method stub }
    @Override
    public void onStatusChanged(String provider, int status,
    Bundle extras) {
        // TODO Auto-generated method stub }
    public void pesan(String key){

```

```
        AlertDialog.Builder error = new
AlertDialog.Builder(this);
        error.setMessage("Pencarian data "+key+"\ntidak
ditemukan");
        error.setNeutralButton("Close", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int
which) {
                // TODO Auto-generated method stub });
            error.show();}}
```